Influence Based Reward Shaping in Multiagent Systems

Doctoral Consortium

Everardo Gonzalez
Oregon State University
Corvallis, Oregon, United States
gonzaeve@oregonstate.edu

ABSTRACT

Learning based approaches work well to coordinate multiagent systems for a broad range of applications. A key challenge in multiagent learning is that the system reward captures the performance of many agents, making it difficult to determine which agents' actions were helpful. Reward shaping helps address this challenge by isolating the direct impact of an agent's actions. However, when an agent's impact is indirect - such as influencing other teammates - then existing approaches struggle. Influence based reward shaping addresses indirect impacts by rewarding an agent based on not just its own actions, but also the actions of agents it influenced. Preliminary results demonstrate that this approach leads to better coordination in a guidance mission where leaders must learn to guide followers to points of interest.

KEYWORDS

Multiagent Systems; Reward Shaping

ACM Reference Format:
Everardo Gonzalez. 2025. Influence Based Reward Shaping in Multiagent Systems: Doctoral Consortium. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 3 pages.

1 INTRODUCTION

Multiagent learning can be used to discover coordinated behaviors in many domains, including warehouse management [5, 15], ocean monitoring [12, 13], and space exploration [17]. A key challenge is that the contributions of every agent are combined in a single system reward. This makes structural credit assignment necessary to determine which agent gets credit for what part of the system reward. Otherwise, each agent tries to optimize a reward that its individual actions have little control over.

Reward shaping helps address credit assignment by isolating an agent's direct impact on the system reward [2, 4, 9, 14, 16]. Difference rewards in particular give each agent a shaped, agent-specific reward that is sensitive only to that agent's actions, and aligned with the system reward [2, 4, 16]. Unfortunately, this shaped reward requires an agent to have a direct impact on the system reward, and in many cases an agent instead has a more subtle, indirect impact. This indirect impact could include clearing a path for its teammates, directing teammates on where to go, or even sharing critical mission information it discovers with teammates.

Influence based reward shaping addresses indirect impacts by giving an agent credit for influencing other agents in the system [10, 11]. The insight here is to think about how agents' interactions affect their decision-making rather than trying to completely isolate an agent's actions for credit assignment. Agents rarely work in isolation; they make decisions based on what they see others doing.

My research is focused on developing shaped rewards for coordinating systems that require influence-based interactions to achieve success. Preliminary results demonstrate how influence based reward shaping can improve coordination in a guidance mission where leaders must learn to guide pre-programmed followers to points of interest. The system reward is based on how close followers get to points of interest, so leaders must learn to influence followers to optimize the system reward.

2 BACKGROUND

The difference reward is a reward shaping method that computes direct credit in a multiagent system and has successfully improved performance in various domains [1–3]. By comparing the system reward to a counterfactual reward with agent i's actions removed, we can isolate agent i's direct impact on system performance. The structure is shown below, where D_i is the difference reward for agent i, $G(z)$ is the system reward with all agents' actions, and $G(z_{-i})$ is the counterfactual reward with agent i's actions removed.

$$D_i = G(z) - G(z_{-i}) \tag{1}$$

3 INFLUENCE BASED REWARD SHAPING

The first step in influence based reward shaping is the indirect difference reward, D-Indirect. This borrows the structure of D, and modifies it in order to compute indirect credit. Rather than removing only agent i for the counterfactual reward, D-Indirect removes agent i as well as other agents that were *influenced* by agent i. The motivating idea is that even if an agent's actions do not directly impact the system reward, that agent can still have important interactions with other agents that we can measure as "influence". This makes it so that if agent i takes actions that set up agent i''s actions to have a direct impact, then agents i and i' both get credit for those actions.

We represent D-Indirect as $D^{Indirect}$, and compute it by taking Equation 1 from the standard difference reward and modifying the counterfactual reward. Rather than removing only agent i, agents in the set F_i are removed. F_i is the set of agent i and agents influenced by agent i. We compute $D^{Indirect}$ according to the following equation.

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

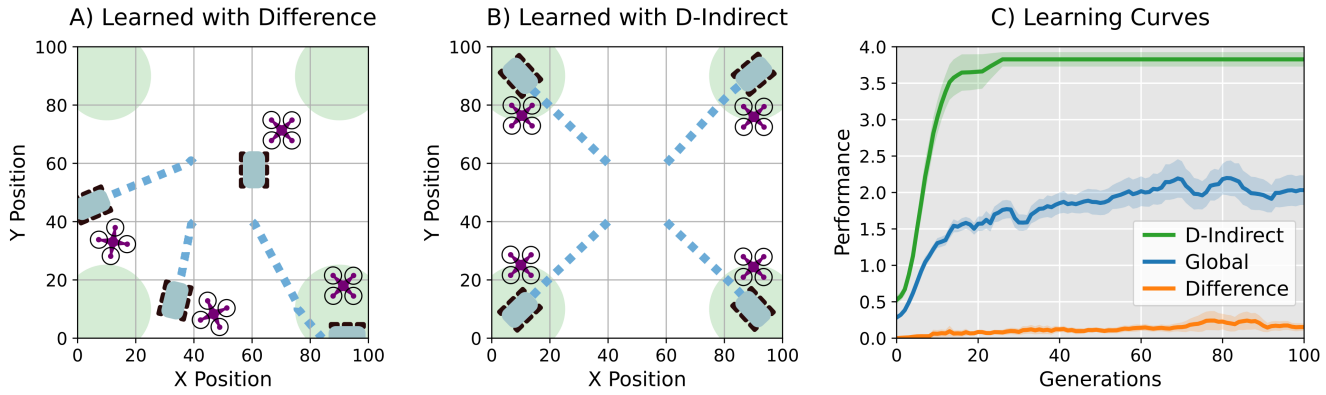

Figure 1: Leaders (purple drones) must guide followers (blue rovers) to POIs (green circles, shading represents capture radius). Dotted lines indicate the path taken by a leader-follower pair after 100 generations of training, with markers indicating final positions. Leaders are slightly offset for easy viewing. A) Leaders struggle to learn with completely uninformative difference rewards. B) Each leader guides its follower to a POI using influence-based indirect difference rewards. C) The only method that reliably gets close to a score of 4.0 (capturing all four POIs) is D-Indirect because this rewards each leader based on how it influenced its follower. D-Indirect provides a clean and informative reward signal for each leader's influence; G and D do not.

$$D_i^{Indirect} = G(z) - G(z_{-F_i} \cup c_{F_i}) \tag{2}$$

F_i is assembled using a domain-specific heuristic that determines which agents are influenced by agent i. In work discussed here, this heuristic is distance-based. If a leader is close to a follower, then that follower is included in that leader's influence set.

4 PRELIMINARY RESULTS

Leaders must learn to guide preprogrammed (non-learning) followers to various points of interest (POIs) in order to maximize G, shown below. G is based on the state of the system after 50 time steps. s_{final} includes the final positions of all followers in the system. The indicator function I returns 1 if a follower is within the capture radius of the POI, and 0 otherwise. The way to get the highest score in G is if each POI is captured by a follower. The leaders learn using a Cooperative Coevolutionary Algorithm (CCEA) [6–8].

$$G(s_{final}) = \sum_j \frac{I(i,j)}{\min_i(dist(follower_i, POI_j))} \tag{3}$$

The map configuration we use for these preliminary results includes 4 leader and follower pairs that start in the center and must navigate to POIs tucked into the corners. Figures 1A and 1B show joint trajectories learned with difference rewards and indirect difference rewards, respectively. We see learning curves (mean and standard error across 5 trials) for different reward shaping techniques in Figure 1C. Difference rewards perform particularly poorly because leaders have no direct impact on the system reward, so no matter what the leaders do, D computes the same reward of zero. Global rewards perform better because leaders are now being rewarded for good behaviors, but unfortunately G is noisy from all the other leaders and followers in the system. It is only when we use D-Indirect that we see the best performance from these agents. D-Indirect rewards a leader based on the actions of any followers it influenced. This makes it so that leaders can learn based on their particular influence in the system. For more extensive explorations of influence based reward shaping, see [10] and [11].

5 PROPOSED RESEARCH

Future research breaks down into two main research objectives.

(1) How to characterize influence between agents in a system when influence is not trivially measured?
(2) How to extend an agent's influence-based reward to capture the network effect of agent influences?

Impact of Achieving Objective 1: How agents interact with each other might not trivially fit into a domain-specific heuristic. The interactions necessary to promote influential behaviors may not even be known apriori. We need to develop *Heuristic-Free Influence* to define fundamentally what influence means in a shaping context. We can borrow concepts from mutual information to measure the causal effect of an agent's action on other agent's actions. This makes it so influence based reward shaping can generalize without requiring domain knowledge.

Impact of Achieving Objective 2: An agent's influence may reach much further than the agents it directly interacted with, and actually propagate throughout an entire network of agents in the system. We need to incorporate *Structural Influence Propagation* where an agent might influence one agent that influences another and so on until an agent in that chain finally has a direct impact. We cannot simply give an agent credit for the direct impact of every agent its influence propagated to because that can quickly become the team reward, putting us back at square one. Instead, we might use a discounted influence reward based on how far removed an agent's influence is from a direct impact that influence is connected to. This makes it so agents can not only learn to directly influence other agents, but more subtly influence an entire network of agents.

REFERENCES

[1] Adrian Agogino and Kagan Tumer. 2004. Efficient Evaluation Functions for Multi-rover Systems. In *Genetic and Evolutionary Computation – GECCO 2004*, Kalyanmoy Deb (Ed.). Springer, Berlin, Heidelberg, 1–11.

[2] A. K. Agogino and K. Tumer. 2008. Analyzing and visualizing multiagent rewards in dynamic and stochastic domains. *Autonomous Agents and Multi-Agent Systems* 17 (2008).

[3] Adrian K Agogino and Kagan Tumer. 2012. A multiagent approach to managing air traffic flow. *Autonomous Agents and Multi-Agent Systems* 24 (2012), 1–25.

[4] Jacopo Castellini, Sam Devlin, Frans A. Oliehoek, and Rahul Savani. 2022. Difference rewards policy gradients. *Neural Computing and Applications* (Nov. 2022). https://doi.org/10.1007/s00521-022-07960-5

[5] Ho-Bin Choi, Ju-Bong Kim, Youn-Hee Han, Se-Won Oh, and Kwihoon Kim. 2022. MARL-Based Cooperative Multi-AGV Control in Warehouse Systems. *IEEE Access* 10 (2022), 100478–100488. https://doi.org/10.1109/ACCESS.2022.3206537

[6] Joshua Cook and Kagan Tumer. 2021. Ad hoc teaming through evolution. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion* (Lille, France) *(GECCO '21)*. Association for Computing Machinery, New York, NY, USA, 89–90. https://doi.org/10.1145/3449726.3459560

[7] Joshua Cook and Kagan Tumer. 2022. Fitness shaping for multiple teams. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Boston, Massachusetts) *(GECCO '22)*. Association for Computing Machinery, New York, NY, USA, 332–340. https://doi.org/10.1145/3512290.3528829

[8] Joshua Cook, Kagan Tumer, and Tristan Scheiner. 2023. Leveraging Fitness Critics To Learn Robust Teamwork. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Lisbon, Portugal) *(GECCO '23)*. Association for Computing Machinery, New York, NY, USA, 429–437. https://doi.org/10.1145/3583131.3590497

[9] Sam Devlin, Logan Yliniemi, Daniel Kudenko, and Kagan Tumer. 2014. Potential-based difference rewards for multiagent reinforcement learning. *Conference on Autonomous Agents and Multi-Agent Systems* (2014).

[10] Everardo Gonzalez, Siddarth Viswanathan, and Kagan Tumer. 2024. Indirect Credit Assignment in a Multiagent System. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems* (Auckland, New Zealand) *(AAMAS '24)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2288–2290.

[11] Everardo Gonzalez, Siddarth Viswanathan, and Kagan Tumer. 2024. Influence Based Fitness Shaping for Coevolutionary Agents. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Melbourne, VIC, Australia) *(GECCO '24)*. Association for Computing Machinery, New York, NY, USA, 322–330. https://doi.org/10.1145/3638529.3654175

[12] Maryam Kouzehgar, Malika Meghjani, and Roland Bouffanais. 2020. Multi-Agent Reinforcement Learning for Dynamic Ocean Monitoring by a Swarm of Buoys. In *Global Oceans 2020: Singapore – U.S. Gulf Coast*. 1–8. https://doi.org/10.1109/IEEECONF38699.2020.9389128

[13] Dohyun Kwon, Joohyung Jeon, Soohyun Park, Joongheon Kim, and Sungrae Cho. 2020. Multiagent DDPG-Based Deep Learning for Smart Ocean Federated Learning IoT Networks. *IEEE Internet of Things Journal* 7, 10 (2020), 9895–9903. https://doi.org/10.1109/JIOT.2020.2988033

[14] Aida Rahmattalabi, Jen Jen Chung, Mitchell Colby, and Kagan Tumer. 2016. D++: Structural credit assignment in tightly coupled multiagent domains. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 4424–4429. https://doi.org/10.1109/IROS.2016.7759651

[15] Oren Salzman and Roni Stern. 2020. Research challenges and opportunities in multi-agent path finding and multi-agent pickup and delivery problems. In *Proceedings of the 19th International Conference on Autonomous Agents and Multi-Agent Systems*. 1711–1715.

[16] David H Wolpert, Kagan Tumer, and Keith Swanson. 2000. Optimal Wonderful Life Utility Functions in Multi-Agent Systems. (2000).

[17] Logan Yliniemi, Adrian K Agogino, and Kagan Tumer. 2014. Multirobot coordination for space exploration. *AI Magazine* 35, 4 (2014), 61–74.